(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 April 2002 (11.04.2002)

PCT

(10) International Publication Number
WO 02/29783 A1

(51) International Patent Classification[7]: G10L 15/14

(21) International Application Number: PCT/CN00/00297

(22) International Filing Date:
30 September 2000 (30.09.2000)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: **INTEL CORPORATION** [US/US]; 2200 Mission College Boulevard, Santa Clara, CA 95052 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **ZHAO, Yibao** [CN/CN]; Yongfengjixie, Xuefu Road, Nangang District, Harbin City, Heilongjiang 150086 (CN). **YAN, Yonghong** [CN/CN]; Apt. 307, Kempinsky Apt., No. 50 Liang Ma Qiao Rd, Chao Yang District, Beijing 100016 (CN). **LIN, Zhiwei** [SG/SG]; BLK518, Jurong West, Street 52, #11-149, Singapore 640518 (SG).

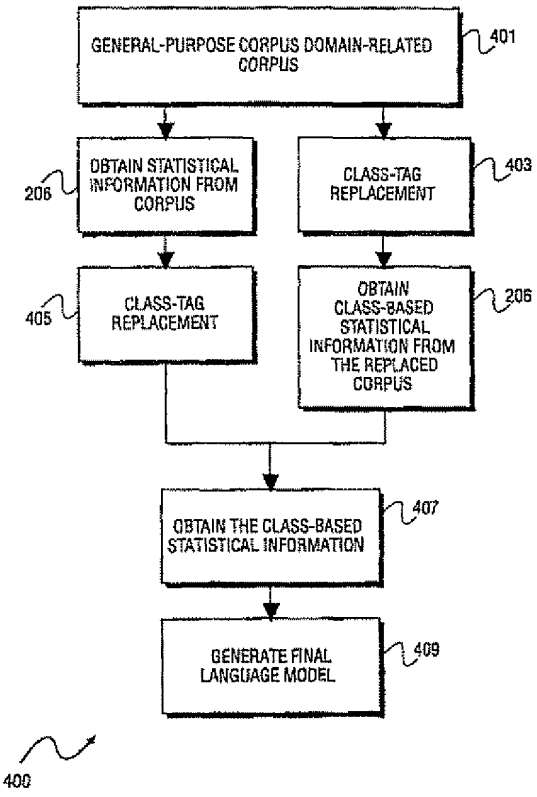
(74) Agent: **CCPIT PATENT AND TRADEMARK LAW OFFICE**; 8th Floor, 2 Fuchengmenwai St., Beijing 100037 (CN).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European

[Continued on next page]

(54) Title: METHOD AND SYSTEM FOR USING RULE-BASED KNOWLEDGE TO BUILD A CLASS-BASED DOMAIN SPECIFIC STATISTICAL LANGUAGE MODEL



(57) Abstract: A method and system for providing a class-based statistical language model representation from rule-based knowledge. The class-based language model is generated from a statistical representation of a class-based rule net. A class-based rule net is generated using the domain-related rules with words replaced with their corresponding class-tags that are manually defined. The class-based statistical representation from the class-based rule net is combined with a class-based statistical representation from a statistical language model to generate a language model. The language model is enhanced by smoothing/adapting with general-purpose and/or domain-related corpus for use as the final language model. A two-pass search algorithm is applied for speech decoding.

WO 02/29783 A1

patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:

— *with international search report*

METHOD AND SYSTEM FOR USING RULE-BASED KNOWLEDGE TO BUILD A CLASS-BASED DOMAIN SPECIFIC STATISTICAL LANGUAGE MODEL

FIELD OF THE INVENTION

The present invention relates generally to speech processing and to speech recognition systems. More particularly, the present invention relates to a method and system for using rule-based knowledge to build a class-based domain specific statistical language model.

BACKGROUND OF THE INVENTION

The ability of computer systems to recognize speech has vastly improved with the progress of technology. These improvements have given rise to new areas of speech recognition technology which are being utilized in various fields today. Language models play a very important role in speech recognition systems. There are two common types of language models that are used today. One is a rule-based statistical language model (RSLM) and the other is a statistical language model (SLM).

SLM is statistics-based and uses a large amount of text to determine its model parameters automatically. The parameters govern the natural language processing or speech recognition in an SLM. An SLM can be trained more easily and decode at a faster speed, however, it has many disadvantages. A disadvantage of using an SLM is that it lacks in quality since it depends on a corpus to train the SLM. A corpus is a data set collected from real-world applications. For example, text from a newspaper is considered as a text corpus. Therefore, the SLM requires a huge corpus with a very large coverage to perform at sufficient levels. In practice, the large corpus and coverage requirements impose great limitations, especially in narrow-domain dialogue system. Thus, building an SLM becomes very difficult and even if build results in poor performance.

A rule-based statistical language model (RSLM) can be used to overcome these drawbacks. The RSLM obtains the statistical information directly from the rule net, and builds a statistical language model with the statistical information. A rule net is a net of grammar rules derived from general linguistics or domain knowledge such as syntactic or semantic knowledge. These rules are created for governing the use of words in the rule net. The disadvantage of RLM is that it works well only in closed

environment. Another disadvantage of using a rule-based system is that the created rules are often not complete enough to cover all circumstances when the system works in an open environment. Due to lack of complete knowledge, the rule-based system lacks the ability to perform accurately and with precision. Another disadvantage of a rule-based system is when a large amount of rules are used decoding speed slows down drastically and creates a fatal situation during real-time system implementation.

BRIEF DESCRIPTION OF THE DRAWINGS

The features and advantages of the present invention are illustrated by way of example and not intended to be limited by the figures of the accompanying drawings in which like references indicate similar elements and in which:

FIG. 1 is a block diagram of an exemplary speech processing system according to one embodiment;

FIG. 2 is a flow chart of an operation to generate a final class-based language model using statistical representations according to one embodiment;

FIG. 3 is a flow chart of an operation to generate a final class-based language model using class-based rule net according to one embodiment;

FIG. 4 is a flow chart of an operation to generate a class-based language model using class-based statistical representations according to one embodiment;

FIG. 5 is an illustration of a rule net and possible paths for the rule net according to one embodiment;

FIG. 6 is a flow chart of an operation to generate a final class-based language model using a statistical process to obtain a statistical representation according to one embodiment;

FIG. 7 is a functional flow diagram of applying smoothing/adapting to a generated class-based RSLM for use as a final class-based language model according to one embodiment;

FIG. 8 an exemplary digital processing system for practicing one embodiment; and

FIG. 9 is a functional flow diagram of applying a two-pass search method to obtain a final result according to one embodiment.

DETAILED DESCRIPTION

A method and system for using rule-based knowledge to build a class-based domain specific statistical language model is described. Words of domain-related rules are replaced with their corresponding class-tags to generate class-based domain-related rules. A class-based rule net is generated using the class-based domain-related rules. A class-based statistical representation is obtained using the class-based rule net from an SLM. The class-based statistical representation from the SLM is combined with a class-based statistical representation from an RLM. A two-pass search decoding method with class-based RSLM is applied.

Replacing words with class-tags prevents a language model from becoming huge in size and drastically slowing down the system performance and makes speech processing task implementation easy. In addition, manually defining classes allows careful class-selection ensuring that words in a class have the same function and semantic meaning maintaining a consistency. Class consistency helps in easy speech processing and uses minimal memory space. Thus, speech-processing systems with class-based language models can operate at faster speeds while maintaining accuracy in speech recognition or natural language processing.

By combining both class-based statistical representations, the language model possesses both real-world and hand-crafted knowledge. This large knowledge base allows a language model to be used effectively in recognizing a variety of domain specific speeches or natural language processing. In addition, by combining both class-based statistical representations, the language model represents adequately both models. Thus, the system can operate at faster speeds while maintaining accuracy in speech recognition or natural language processing.

Conventional class-based decode search method calculates the sentence probability through multiplying the probability of sentence clause and the probability of words of each class in the sentence clause together. It's drawback is when the item number of each class changes (for example, the number of the names in class name changes, this is often happened in real-world application), the language model should be retrained. Two-pass search method in this invention separates the calculation into two steps, which can overcome the drawbacks.

FIG. 1 is a block diagram of an exemplary speech processing system 150 for practicing the invention. The speech processing system 150 is able to process speech

into a text format. The speech processing system 150 includes a microphone 154, a feature extractor 156 coupled to the microphone 154, an acoustic model 158 coupled to the feature extractor 156, and a language model 160 coupled to the acoustic model 158. Items of the speech processing system 150 could be presented as circuitry or software.

A feature extractor 156 coupled to a microphone 154 receives a speech signal from the microphone 154. The speech signal corresponds to speech by a user represented by reference number 152 spoken into the microphone 154. The feature extractor 156 receives the speech signal and converts the spectral measurements of speech to a set of features. The set of features describe the acoustic properties of various phonetic sounds for example nasal or friction sounds.

An acoustic model 158 coupled to the feature extractor 156 receives the set of features. The acoustic model 158 compares the set of features to acoustic-phonetic sounds stored in the model. The acoustic model 158 decodes the set of features according to stored acoustic-phonetic sounds and sends the decoded information to the language model 160.

A language model 160 coupled to the acoustic model 158 receives information from the acoustic model 158. The information contains decoded information from the acoustic model as well as information that has been sent by the feature extractor 156, which has not been decoded by the acoustic model 158. The language model 160 evaluates both decoded and non-decoded information and converts to text. The language model building method is further described in **FIG. 2.**

FIG. 2 is a flow chart of an operation 200 to generate a final language model using a statistical representation according to one embodiment. Operation 200 can be implemented by digital processing system 800. Operation 200 also allows an algorithm to generate a class-based rule net using domain-related class-based rules and use the rule net to obtain a statistical representation. The statistical representation is used for generating a final language model for a speech processing system.

Referring to **FIG. 2**, at operation block 201, words of domain related rules are replaced with their corresponding class-tags, and class-based domain-related rules are obtained. **FIG. 3** and 6 further describes detailed aspects of obtaining class-based domain related rules and their use for various embodiments.

At operation block 203, a class-based rule net is generated. The rule net is composed of rules. Rules are created for governing the use of words in the rule net.

FIGS. 3 and 6 further describe detailed aspects of collecting rules to generate rule nets and rule creations for various embodiments.

At operation block 205, a statistical representation of the class-based rule net is obtained. Class-based rule net is a set of hand-crafted grammar rules derived from general linguistics or domain knowledge such as syntactic or semantic knowledge. The words of the grammar rules are replaced with their corresponding class-tags to generate a class-based rule net. These rules are generated for governing the use of words in the rule net.

A statistical representation of the class-based rule net is obtained from the generated class-based rule net. The statistical representation is obtained through traversing the class-based rule net or from the sentences generated from the rule net. A statistical formula is applied to obtain the statistical representation while traversing the rule net. **FIG. 6** further describes detail aspects of obtaining a statistical representation of the class-based rule net.

At operation block 202, general-purpose or domain-related corpus are obtained. The statistical language model obtained from an RLM is often not robust. A general-purpose or domain-related corpus can be used to smooth the RLM-based SLM.

At operation block 204, the corpus is preprocessed. The corpus is obtained from real-world practice and therefore has no class-tag when obtained. The statistical representation obtained from the class-based RSLM has class-tags. The corpus is preprocessed to make it compatible with the class-based RSLM and used to smooth the class-based RSLM. **FIG. 4** further describes detail aspects of preprocessing.

At operation block 206, a class-based statistical language model is built from the preprocessed corpus obtained in block 204. As described previously, the statistical language model uses a large amount of text of corpus to determine automatically the language model parameters. A class-based statistical representation of the class-based statistical language model is obtained. The statistical language model is further described in **FIG. 4**.

At operation block 207, class-based statistical representations of the class-based rule-based and class-based statistical language models are combined. For example, a computer having a predefined algorithm may combine the class-based statistical representations using a predefined algorithm. The combination is often enhanced by smoothing/adapting the rule-based SLM with a general-purpose SLM or domain-

related SLM. **FIG. 7.** further describes detail enhancing aspects using smoothing /adapting.

At operation block 208, a final language model is generated. The final language model is a result of combining the class-based statistical representation of class-based RSLM and class-based statistical language model.

FIG. 3 is a flow chart of an operation 300 to generate a final language model using a class-based rule net according to one embodiment. Operation 300 also allows generation of a final language model to be used in a speech processing system by defining a class and replacing words that belong to the class with class-tags to generate a class-based rule net and obtain a statistical representation.

Referring to **FIG. 3**, at operation block 301 domain related rules are obtained for use in generating a rule net. Domain related rules are handcrafted grammar rules. Preparation of domain related rules require obtaining as many domain related rules as possible. Alternatively, preparation of domain related rules require obtaining domain related rules as broad as possible. Numerous and broad domain related rules aid in providing enough knowledge for generation of a good performance rule net.

For example, domain related rules may be written using rule-based grammar which complies with Backus-Naur Form (BNF) which is a standard grammar form as follows:

$tagname = Zhiwei Lin | Jian Han | Yibao Zhao;
$attendee = $tagname [$attendee];
$room = room A | room B;
$rule1 = I want to have a meeting with $attendee;
$rule2 = I would like to meet with $attendee today;
$rule3 = I would like to have a meeting in $room today;
$rule4 = I want to have a meeting in $room.

In the above example, the symbol "|" indicates "or", the symbol "[x]" indicates that "x" can be omitted, and the symbol "$x" indicates that it is a variable. The above exemplary rules can be used to derive a large number of sentences. For example, using the above exemplary rules the following sentences can be derived:

I want to have a meeting with Zhiwei Lin.
I want to have a meeting with Zhiwei Lin, Jian Han.
I want to have a meeting with Yibao Zhao, Zhiwei Lin.
I would like to meet with Jian Han, Zhiwei Lin today.
I want to have a meeting in room A.
I would like to have a meeting in room B today.

At operation block 303, classes are defined manually. A class is a word set that contains words with the same or similar function. Classes are manually defined so that they can be carefully selected to ensure that words in a class have the same or similar function or semantic meaning. This careful selection aids in language understanding. For example, all person names can be considered as a class. Therefore, as discussed earlier in the above example, names such as Zhiwei Lin, Jian Han and Yibao Zhao can be considered as a class with a class-tag name "name", and room information such as room A, room B can be considered as a class with a class-tag name "room".

At operation block 305, words in each rule that belong to a defined class are replaced with their corresponding class-tags. For example, words can be replaced in the above example with class-tags as follows:

$tagname = <name>;
$attendee = $tagname [$attendee];
$tagname2 = <room>;
$room = $tagname2 [$room];
$ww = want | I would like;
$rule1 = I $ww to have a meeting with $attendee [today];
$rule2 = I $ww to meet with $attendee [today];
$rule3 = I $ww to have a meeting in $room [today];
$rule4 = I $ww to have a meeting in $room.

where <name> = {Zhiwei Lin, Jian Han, Yibao Zhao}, <room> = {room A, room B} are class-tags, "[x]" means "x" can be omitted, "$x" means it is a variable.

At operation block 203, a class-based rule net is generated using the domain-related rules with words replaced by class-tags. An exemplary rule net is shown in FIG. 5 in which each path from a beginning to an end expresses a valid sentence.

At operation block 205, statistical information is obtained. As described previously, the statistical information is obtained from the rule net with class-tags. The obtained statistical information is used for generating a final language model. Smoothing/adapting may be applied before using the language model as a final language model.

FIG. 4 is a flow chart of an operation 400 to generate a class-based language model using class-based statistical representations according to one embodiment. Operation 400 also describes methods to build a class-based SLM from a general-purpose or domain-related corpus.

Referring to **FIG. 4**, at operation block 401, general-purpose or domain-related corpus is obtained. The class-based SLM obtained from a RLM is often not robust. A general-purpose or domain-related corpus can be used to smooth the RLM-based SLM.

At operation block 403, words are replaced with their corresponding class-tags and applied to the general-purpose or domain-related corpus. Since the corpus is obtained from real-world application, it does not contain class-tags when obtained. Therefore, the corpus is preprocessed to make it compatible with the class-based RSLM. Preprocessing consists of replacing the words of the corpus with their corresponding class-tags. The replacement is performed on each word of the corpus. The replacement is applied according to the following three rules:

> Rule 1. If a word has steady function and meaning in different context (it often appears in only one class). Then the word can be replaced directly with it's corresponding class-tag; for example, a person name always has the same function, expresses the same meaning even in a different context, so all the person names in the corpus can be replaced with person name's class-tag.
> Rule 2. If a word shows different function and expresses different meaning in different contexts (it often appears in more than one class). Then the word should be replaced according to its context. This is performed by comparing the word's context, if the context is same, the word should be replaced with the corresponding class-tag. For example, a word "one" in a corpus may belong to <hour> class or a <minute> class. If the word's context in the corpus is "one hour", and there exists a class for "<hour> hour", then the word should be replaced with class-tag <hour>. If the word's context in corpus is "one minute", and there exists a class for "<minute> minute", then the word should be replaced with class-tag <minute>.
> Rule 3. If Rule 1 and Rule 2 are not satisfied then the word shall not be replaced with a class-tag.

At operation block 405, class-tag replacement is applied to the statistical information obtained in operation block 206. Alternatively, class-tag replacement is applied directly to the corpus as described previously in operation block 403.

At operation block 407, class-based statistical information is obtained. The class-based statistical information is used to build a class-based statistical language model.

At operation block 409, as previously discussed previously, a final class-based statistical language model is generated using the class-based statistical information.

FIG. 6 is a flow chart of an operation 600 to generate a final language model using a statistical process to obtain class-based statistical representations according to one embodiment. Operation 600 also allows generation of a final language model by training a class-based statistical language model directly from a class-based rule net that is generated using domain related rules.

Referring to **FIG. 6**, at operation block 603, a statistical process is applied to the class-based rule net for obtaining class-based statistical representation. The statistical process may be performed by using a computer (e.g., digital processing system 800) using an algorithm that describes the training process. The statistical process is to go through each path of a class-based rule net from a beginning word to an ending word. **FIG. 5** shows an exemplary rule net in which the arrows point to all the possible paths that can be taken. The statistical process goes through each possible path and obtains a statistical representation.

At operation block 605, a calculation is performed. The calculation consists of counting the frequency of each word sequence in the rule net. A class-tag is regarded as a word in the rule net. Therefore, a word sequence could either a word or a class-tag. An algorithm is executed as part of the statistical process to review all the word sequences in the rule net and follows each path in the rule net for counting the frequency of each word sequence in a path. The word sequences at same position in the rule net are counted only once although they may appear in more than one path.

The counting process is performed by using a formula (1) shown below. An algorithm executes the formula (1) to follow each path in the class-based rule net and count the same part in the rule net only once. In operation, the frequency of the word sequence is dictated by the following formula (1).

$$C(W_i, W_{i+1}, \cdots, W_{i+N-1}) = \sum_{\substack{W_i = W_{j_{(i-1)+1}}, \; l = i, \cdots, i+N-1 \\ (W_{j_1}, W_{j_2}, \cdots, W_{j_N}) \in PNet}} (W_{j_1}, W_{j_2}, \cdots, W_{j_N}) \quad (1)$$

In the above formula, "Pnet" is the N-words-length valid part path set of the net. An "N" is the history length in Ngram, and j_i is the index of each word in the net. For example, when applying formula (1) to the rule net, the following can be obtained:

C(I, want) = 1
C(want, to) = 1
...
C(with, <NAME>) = 1
...

Where C(with, <NAME>) =1 although the word pair (with, <NAME>) appears in more than six possible paths in the rule net but it is counted only once. The statistical information obtained by applying the frequency counting formula is used for generating a final language model.

At operation block 607, the class-based statistical information is obtained by applying the counting process from operational block 605. The class-based statistical information is used to generate a class-based statistical language model.

FIG. 7 is a functional flow diagram 700 of applying smoothing/adapting to a generated class-based RSLM for use as a final language model according to one embodiment. Operation 700 also allows smoothing/adapting to be applied to a class-based RSLM before using the language model as a final language model. Operation blocks in **FIGS. 3, 4** and **6** as discussed previously are used in the smoothing operation.

Referring to **FIG. 7**, at operation block 701, a class-based RSLM is generated using operation blocks in **FIGS. 3** and **6**. The language model is similar to the final language model discussed earlier **FIGS. 3** and **6**, however further enhancing operations are performed on the language model before its use as a final language model. Further enhancing functions such as smoothing/adapting, discussed below, aid in obtaining a high quality performance language model.

At operation block 703, domain-related corpus is obtained for smoothing /adapting the class-based RSLM. Domain-related corpus is a set of domain specific data.

At operation block 705, domain-related corpus is pre-processed. The domain-related corpus is pre-processed by replacing the words in the corpus with class-tags. As described previously, a statistical process is applied to the replaced corpus to obtain class-based statistical information. Alternatively, the domain-related corpus is pre-processed by applying the statistical process to the corpus and replacing the words in the statistical information with class-tags.

At operation block 707, a class-based statistical language model is built using the class-based statistical information. The statistical language model is used to smooth/adapt the class-based RSLM.

At operation block 709, general-purpose corpus is obtained for smoothing /adapting the class-based RSLM. General-purpose corpus is real-world data set

collected from real-world applications. For example, text from a newspaper, is considered as a real-world corpus.

At operation block 711, the general-purpose corpus is pre-processed. As described previously preprocessing consists of replacing words in the general-purpose corpus with corresponding class-tags. Preprocessing is performed to make the general-purpose corpus compatible with class-based language model.

At operation block 713, a class-based statistical language model is built using the class-based statistical information. The statistical language model is used to smooth/adapt the class-based RSLM.

At operation block 715, domain-related language model from operation 707 is used in smoothing/adapting the class-based language model from operation 701. Smoothing/adapting with the domain-related language model helps the class-based RSLM to be used effectively in recognizing domain specific speech. Alternatively, a general-purpose language model from operation 713 can also be used to smooth/adapt the class-based language model from operation 701. The real-world corpus used in the general-purpose language model allows the class-based language model to be used effectively in recognizing a variety of speeches that is real-world. Smoothing/adapting the class-based RSLM with the general-purpose SLM also helps cure language indescribable scenario and obtain a robust system performance.

The class-based RSLM is generated from the rules hand-crafted manually, and contains only domain related knowledge. Domain-related knowledge does not cover all of the real-world language phenomena in the domain. Therefore a statistical language model generated from a large amount of text corpus containing rich language phenomena is obtained and applied. The statistical language model covers user-spoken sentences in stochastic manner and aids in covering real-world language.

The general-purpose SLM from operation 707 and/or domain-related SLM from operation 713 are applied separately to the class-based RSLM to create a final language model. Alternatively, both general-purpose SLM from operation 707 and domain-related SLM from operation 713 are applied simultaneously to the generated language model to create a final language model.

Smoothing/adapting enhances the system capability of recognizing voice. Smoothing /adapting is applied by interpolating the parameters of a system with those of a general-purpose statistical language model.

At operation 717, a final language model is generated. The final language model consists of the class-based language model that has been applied with smoothing/adapting.

FIG. 8 is an exemplary digital processing system 800 in which the present invention can be practiced. The speech processing and speech recognition techniques described herein can be implemented and utilized within digital processing system 800, which can represent a general purpose computer, portable computer, hand-held electronic device, or other like device. The components of digital processing system 800 are exemplary in which one or more components can be omitted or added. For example, one or more memory devices can be utilized for digital processing system 800.

Referring to **FIG. 8**, digital processing system 800 includes a central processing unit 802 and a signal processor 803 coupled to a display circuit 805, main memory 804, static memory 806, and mass storage device 807 via bus 801. Digital processing system 800 can also be coupled to a display 821, keypad input 822, cursor control 823, hard copy device 824, input/output (I/O) devices 825, and audio/speech device 826 via bus 801.

Bus 801 is a standard system bus for communicating information and signals. CPU 802 and signal processor 803 are processing units for digital processing system 800. CPU 802 or signal processor 803 or both can be used to process information and/or signals for digital processing system 800. Signal processor 803 can be used to process speech or audio information and signals for speech processing and recognition. Alternatively, CPU 802 can be used to process speech or audio information and signals for speech processing or recognition. CPU 802 includes a control unit 831, an arithmetic logic unit (ALU) 832, and several registers 833, which are used to process information and signals. Signal processor 803 can also include similar components as CPU 802.

Main memory 804 can be, e.g., a random access memory (RAM) or some other dynamic storage device, for storing information or instructions (program code), which are used by CPU 802 or signal processor 803. For example, main memory 804 may store speech or audio information and instructions to be executed by signal processor 803 to process the speech or audio information. Main memory 804 may also store temporary variables or other intermediate information during execution of

instructions by CPU 802 or signal processor 803. Static memory 806, can be, e.g., a read only memory (ROM) and/or other static storage devices, for storing information or instructions, which can also be used by CPU 802 or signal processor 803. Mass storage device 807 can be, e.g., a hard or floppy disk drive or optical disk drive, for storing information or instructions for digital processing system 800.

Display 821 can be, e.g., a cathode ray tube (CRT) or liquid crystal display (LCD). Display device 821 displays information or graphics to a user. Digital processing system 801 can interface with display 821 via display circuit 805. Keypad input 822 is a alphanumeric input device for communicating information and command selections to digital processing system 800. Cursor control 823 can be, e.g., a mouse, a trackball, or cursor direction keys, for controlling movement of an object on display 821. Hard copy device 824 can be, e.g., a laser printer, for printing information on paper, film, or some other like medium. A number of input/output devices 825 can be coupled to digital processing system 800. For example, a speaker can be coupled to digital processing system 800. Audio/speech device 826 can be, e.g., a microphone with an analog to digital converter, for capturing sounds of speech in an analog form and transforming the sounds into digital form, which can be used by signal processor 803 and/or CPU 802, for speech processing or recognition.

The speech processing techniques described herein can be implemented by executing code or instructions stored in a machine-readable medium coupled with or contained in digital processing system 800. For example, CPU 802 or signal processor 203 can execute the code or instructions stored in main memory 804 to process speech or to recognize continuous speech. The machine-readable medium can include a mechanism that provides (i.e., stores and/or transmits) information in a form readable by a machine such as computer or digital processing device. For example, a machine-readable medium may include a read only memory (ROM), random access memory (RAM), magnetic disk storage media, optical storage media, flash memory devices. The code or instructions can be represented by carrier wave signals, infrared signals, digital signals, and by other like signals.

FIG. 9 is a functional flow diagram 900 of applying a two-pass search method to obtain a final result according to one embodiment. The two pass search method is applied to information decoded from the acoustic model 158, and information sent by feature extractor 156. Operation 900 also applies a first pass search method to obtain a

sentence clause and a second pass search method to the sentence clause obtained in the first pass search to obtain a final result.

At operation block 901, the decoded information and information from feature extractor 156 is obtained. Decoded information is information sent from acoustic model 158. The obtained decoded information and information from feature extractor 156 is used as input for the two pass search method.

At operation block 903, a first pass search method is applied. A formula is used in applying the first pass search method to the decoded information. The first pass search method determines all the sentence clauses that succeed from all possible sentences in the language model. . The first pass search matches the all the information obtained with best-matched sentence clauses and obtains a sentence clause. The first pass search applies a formula for determining the match. The formula applied is as follows:

$$\Pr(S) = \prod \Pr(C_i \mid C_{i-N+1}, \ldots, C_{i-1})$$

Where Pr(S) is the probability of sentence clause S and

C expresses class-tag, Ci is the ith class-tag in the sentence clause.

At operation 905, a second pass search method is applied for obtaining a final result. The second pass search is applied to the sentence clauses obtained in the first pass search method. Second pass search is also applied on the decoded information and information from feature extractor 156 simultaneously. Second pass search method matches all the information obtained with the words of each class in the sentence clause obtained in the first pass search, and finds the best-matched word for each class. The second pass search applies a formula for determining the match. The formula applied is as follows:

$$\Pr(S) = \prod_i \Pr(W_j \mid C_i)$$

Where Wj is the jth word in the ith class_of the sentence clause.

At operation 907, the final result is obtained. The result is a text sentence that matches speech obtained by the user 152 accurately using the statistical language model.

Thus, a method and system for providing a statistical representation from rule-based grammar specifications have been described. In the foregoing specification, the invention has been described with reference to specific exemplary embodiments

thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and drawings are, accordingly, to be regarded in an illustrative sense rather than a restrictive sense.

CLAIMS

What is claimed is:

1. A speech processing method comprising:

replacing words of domain-related rules with corresponding class-tags to generate class-based domain-related rules;

generating a class-based rule net using the class-based domain-related rules;

obtaining a class-based statistical representation from the class-based rule net; and

combining the class-based statistical representation with a class-based statistical representation from a statistical language model to generate a final language model.

2. The speech processing method of claim 1, wherein replacing words further includes:

selecting words with same function and semantic meaning;

defining a class manually for the selected words; and

associating each class with a class-tag to replace words of the domain-related rules.

3. The speech processing method of claim 1, further includes applying a two-pass search method for decoding speech with class-based rule-based statistical language model.

4. The speech processing method of claim 1, wherein combining the class-based statistical representations further includes:

using domain-related and/or general-purpose corpus;

preprocessing the domain-related and/or general-purpose corpus to replace words of the words in the domain-related and/or general-purpose corpus with corresponding class-tags to generate a class-based corpus;

obtaining a class-based statistical representation from the class-based corpus; and

generating a rule net using the class-based statistical representation to obtain a statistical language model representation.

5. The speech processing method of claim 1, further comprising applying smoothing/adapting to the final language model before using as a final language model.

6. A system comprising:

a memory to store a statistical language model representation of a class-tag based rule net; and

a processor to replace words of domain-related rules with corresponding class-tags to generate class-based domain-related rules, to generate a class-based rule net using the class-based domain-related rules, to obtain a class-based statistical representation from the class-based rule net, and to combine the class-based statistical representation with a class-based statistical representation from a statistical language model to generate a final language model.

7. The system of claim 6, wherein the processor is to replace the words with class-tags, select words with same function and semantic meaning, to define a class manually for the selected words, and to associate each class with a class-tag to replace words of the domain-relates rules.

8. The system of claim 6, further includes applying a two-pass search method for decoding speech with class-based rule-based statistical language model.

9. The system of claim 6, wherein the processor is to combine the class-based statistical representations, to use domain-related and/or general-purpose corpus, to preprocess the domain-related and/or general-purpose corpus to replace words of the words in the domain-related and/or general-purpose corpus with corresponding class-tags to generate a class-based corpus, to obtain a class-based statistical representation from the class-based corpus, and to generate a rule net using the class-based statistical representation to obtain a statistical language model representation.

10. The system of claim 6, wherein the processor is to apply smoothing/adapting to the final language model before using as a final language model.

11. A storage medium having stored therein a plurality of machine executable instructions, wherein when executed, the instructions perform a method comprising:

replacing words of domain-related rules with corresponding class-tags to generate class-based domain-related rules;

generating a class-based rule net using the class-based domain-related rules;

obtaining a class-based statistical representation from the class-based rule net; and

combining the class-based statistical representation with a class-based statistical representation from a statistical language model to generate a final language model.

12. The storage medium of claim 11, wherein replacing words further includes:

selecting words with same function and semantic meaning;

defining a class manually for the selected words; and

associating each class with a class-tag to replace words of the domain-related rules.

13. The storage medium of claim 11, further includes applying a two-pass search method for decoding speech with class-based rule-based statistical language model.

14. The storage medium of claim 11,further includes applying a two-pass search method for decoding speech with class-based rule-based statistical language model.

15. A speech processing system comprising:

obtaining a sentence clause from a rule net using a first pass search process; and

obtaining a final result from obtained sentence clause using a second pass search process.

16. The speech processing system of claim 15, wherein the first pass and the second pass search process further includes using a formula, wherein the formula for the first pass being

$$\Pr(S) = \prod \Pr(C_i \mid C_{i-N+1}, \ldots, C_{i-1}),$$

and the formula for the second pass being

$$\Pr(S) = \prod_i \Pr(W_j \mid C_i).$$

17. The speech processing system of claim 15, wherein the first pass search process further includes:

determining if a word belongs to a predefined class;

associating the determined word with the predefined class; and

creating a new class for the word that does not belong to a predefined class.

18. The speech processing system of claim 15, further comprising:

generating a class-based rule net;

building a class-based language model using the class-based rule net; and

applying smoothing/adapting to the class-based language model to generate a final class-based language model.

19. The speech processing system of claim 18, wherein applying smoothing/adapting function further includes:

using a statistical language model containing domain-related and/or general-purpose corpus;

pre-processing the domain-related and/or general-purpose corpus to replace words in the domain-related and/or general-purpose corpus with class-tags; and

building a domain-related and/or general-purpose language model from the pre-processed domain-related and/or general-purpose corpus to apply to the class-based language model.

20. A system comprising:

a memory to store a sentence clause and a final result; and

a processor to obtain a sentence clause from a rule net using a first pass search method, and to obtain a final result from obtained sentence clause using a second pass search method.

21. The system of claim 20, wherein the processor is to apply the first pass and second pass search method using a formula, wherein the formula for the first pass being

$$\Pr(S) = \prod \Pr(C_i \mid C_{i-N+1}, \ldots, C_{i-1})$$

and the formula for the second pass being

$$\Pr(S) = \prod_i \Pr(W_j \mid C_i).$$

22. The system of claim 20, wherein the processor is to apply the first pass search process by determining if a word belongs to a predefined class, to associate the determined word with the predefined class, and to create a new class for the word that does not belong to a predefined class.

23. The system of claim 20, wherein the processor is to generate a class-based rule net, to build a class-based language model using the class-based rule net, and to apply smoothing/adapting to the class-based language model to generate a final class-based language model.

24. The system of claim 23, wherein the processor to apply smoothing/adapting further includes the processor to use a statistical language model containing domain-related and/or general-purpose corpus, to pre-process the domain-related and/or general-purpose corpus to replace words in the domain-related and/or general-purpose corpus with class-tags, and to build a domain-related and/or general-purpose language model from the pre-processed domain-related and/or general-purpose corpus to apply to the class-based language model.

25. A storage medium having stored therein a plurality of machine executable instructions, wherein when executed, the instructions perform a method comprising:

obtaining a sentence clause from a rule net using a first pass search process; and

obtaining a final result from obtained sentence clause using a second pass search method.

26. The storage medium of claim 25, further comprising:

generating a class-based rule net;

building a class-based language model using the class-based rule net; and

applying smoothing/adapting to the class-based language model to generate a final class-based language model.

27. The storage medium of claim 25, wherein the first pass search process further includes:

determining if a word belongs to a predefined class;

associating the determined word with the predefined class; and

creating a new class for a word that does not belong to a predefined class.

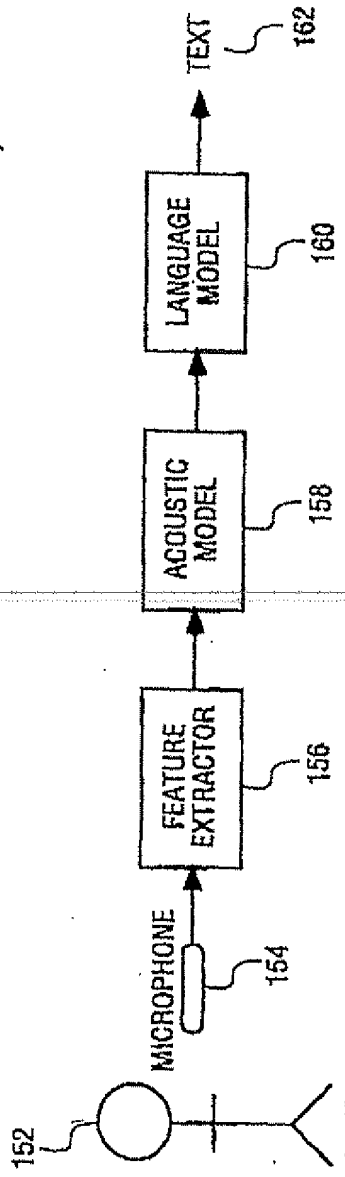

150
TEXT
162
LANGUAGE MODEL
160
ACOUSTIC MODEL
158
FEATURE EXTRACTOR
156
MICROPHONE
154
152
USER

FIG. 1

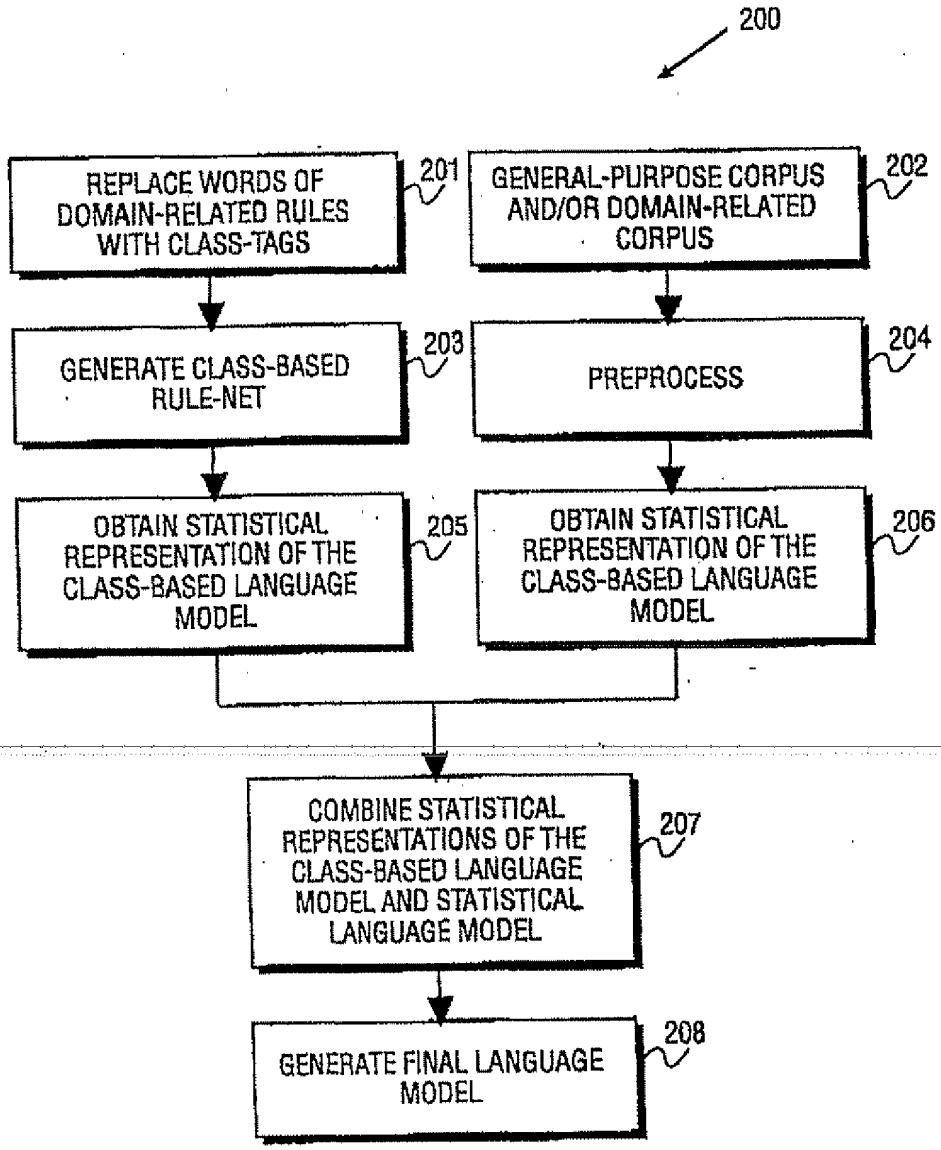

200
REPLACE WORDS OF DOMAIN-RELATED RULES WITH CLASS-TAGS
201
GENERAL-PURPOSE CORPUS AND/OR DOMAIN-RELATED CORPUS
202
GENERATE CLASS-BASED RULE-NET
203
PREPROCESS
204
OBTAIN STATISTICAL REPRESENTATION OF THE CLASS-BASED LANGUAGE MODEL
205
OBTAIN STATISTICAL REPRESENTATION OF THE CLASS-BASED LANGUAGE MODEL
206
COMBINE STATISTICAL REPRESENTATIONS OF THE CLASS-BASED LANGUAGE MODEL AND STATISTICAL LANGUAGE MODEL
207
GENERATE FINAL LANGUAGE MODEL
208

FIG. 2

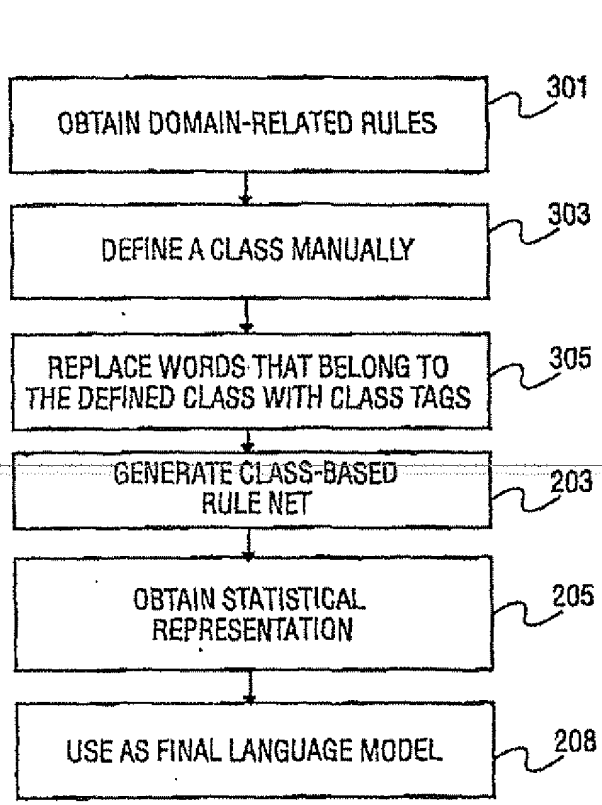

300
OBTAIN DOMAIN-RELATED RULES
301
DEFINE A CLASS MANUALLY
303
REPLACE WORDS THAT BELONG TO THE DEFINED CLASS WITH CLASS TAGS
305
GENERATE CLASS-BASED RULE NET
203
OBTAIN STATISTICAL REPRESENTATION
205
USE AS FINAL LANGUAGE MODEL
208

FIG. 3

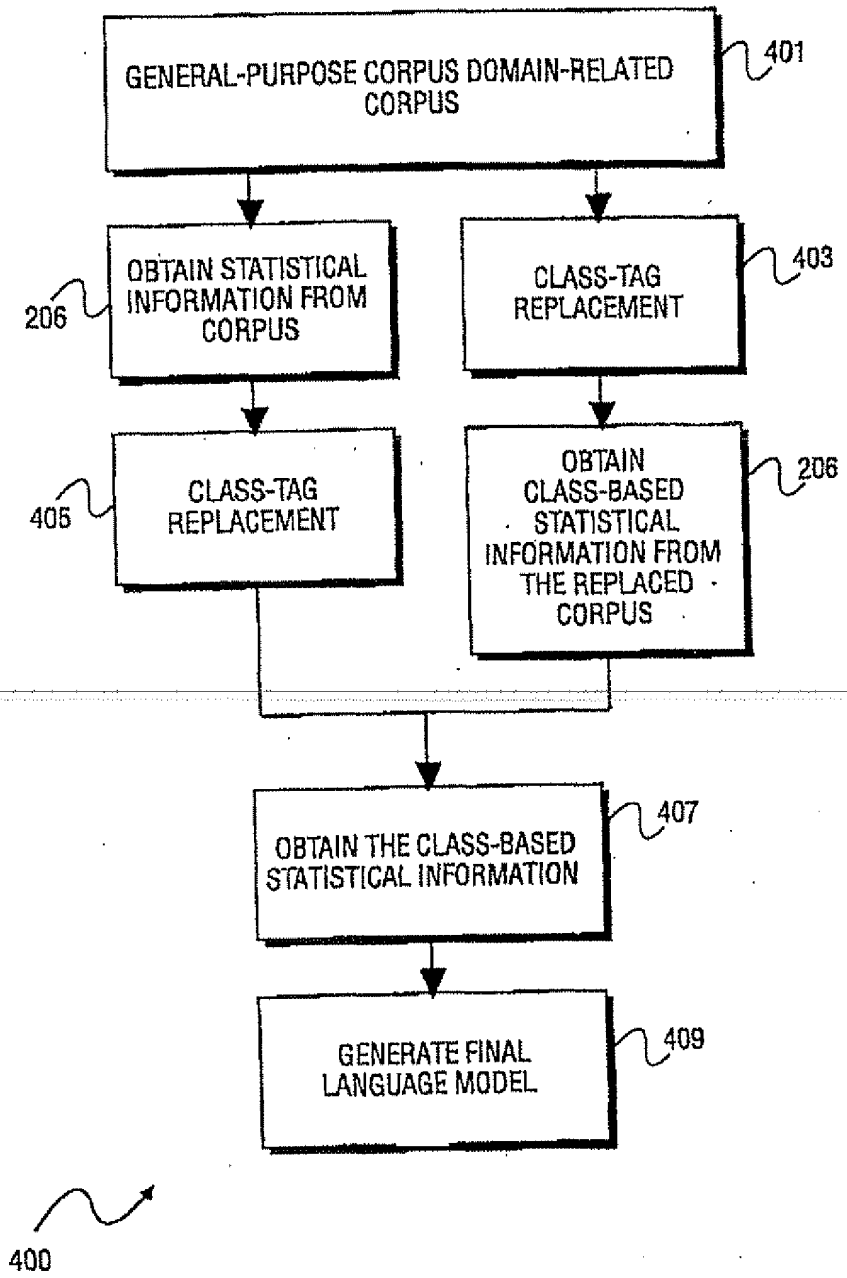

GENERAL-PURPOSE CORPUS DOMAIN-RELATED CORPUS
401
OBTAIN STATISTICAL INFORMATION FROM CORPUS
206
CLASS-TAG REPLACEMENT
403
CLASS-TAG REPLACEMENT
405
OBTAIN CLASS-BASED STATISTICAL INFORMATION FROM THE REPLACED CORPUS
206
OBTAIN THE CLASS-BASED STATISTICAL INFORMATION
407
GENERATE FINAL LANGUAGE MODEL
409
400

FIG. 4

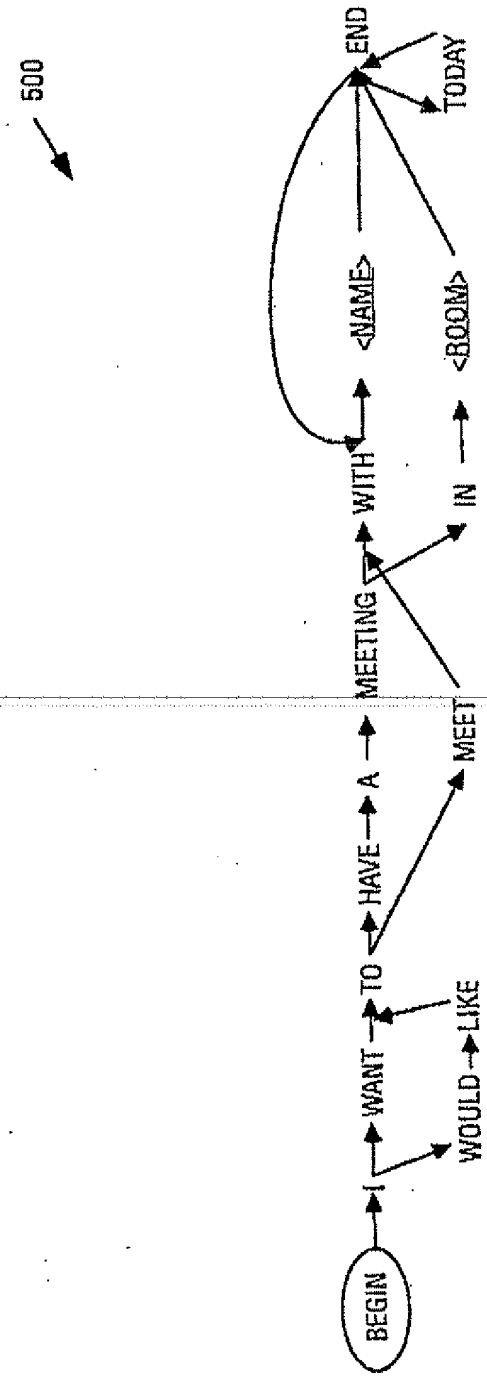

500
BEGIN
I
WANT
WOULD
LIKE
TO
HAVE
A
MEETING
MEET
WITH
IN
<NAME>
<ROOM>
END
TODAY

FIG. 5

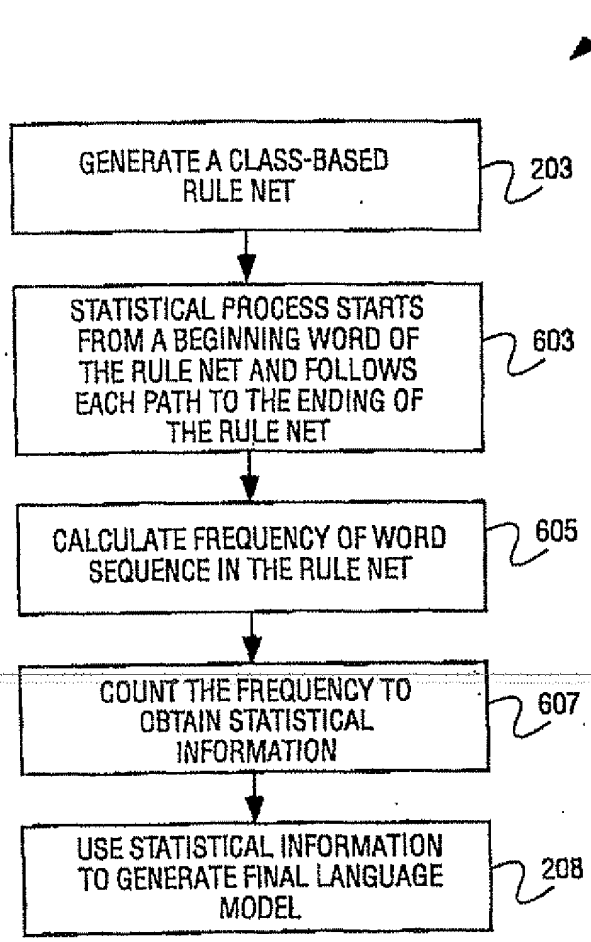

600
GENERATE A CLASS-BASED RULE NET
203
STATISTICAL PROCESS STARTS FROM A BEGINNING WORD OF THE RULE NET AND FOLLOWS EACH PATH TO THE ENDING OF THE RULE NET
603
CALCULATE FREQUENCY OF WORD SEQUENCE IN THE RULE NET
605
COUNT THE FREQUENCY TO OBTAIN STATISTICAL INFORMATION
607
USE STATISTICAL INFORMATION TO GENERATE FINAL LANGUAGE MODEL
208

FIG. 6

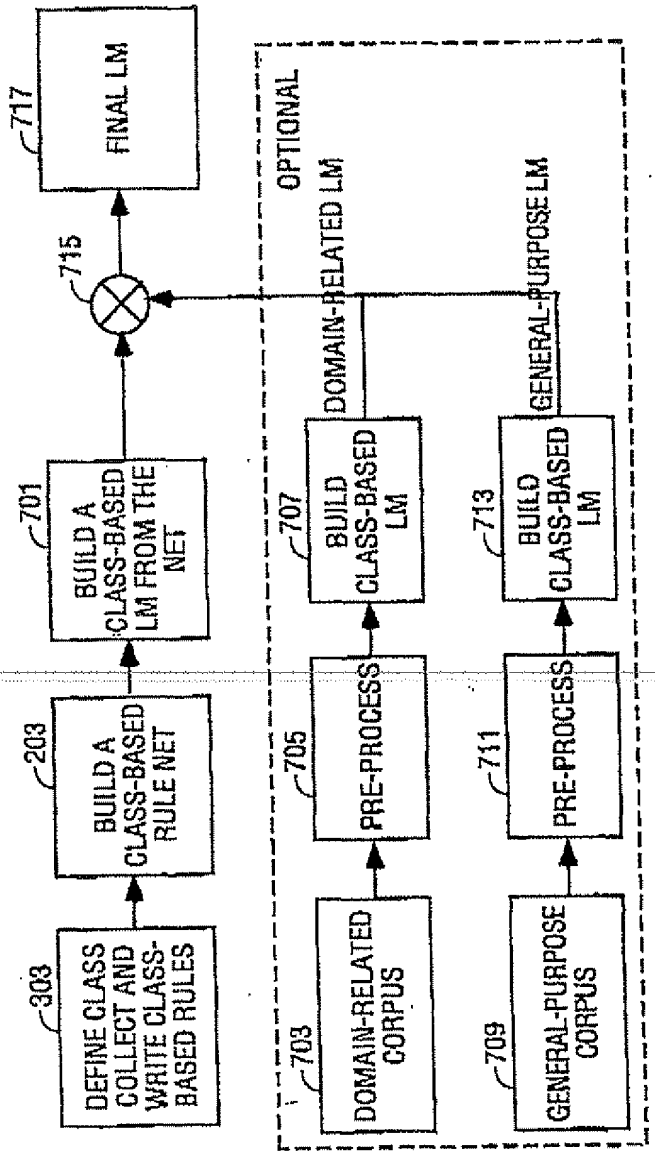

303
DEFINE CLASS COLLECT AND WRITE CLASS-BASED RULES
203
BUILD A CLASS-BASED RULE NET
701
BUILD A CLASS-BASED LM FROM THE NET
715
717
FINAL LM
OPTIONAL
703
DOMAIN-RELATED CORPUS
705
PRE-PROCESS
707
BUILD CLASS-BASED LM
DOMAIN-RELATED LM
709
GENERAL-PURPOSE CORPUS
711
PRE-PROCESS
713
BUILD CLASS-BASED LM
GENERAL-PURPOSE LM

FIG. 7

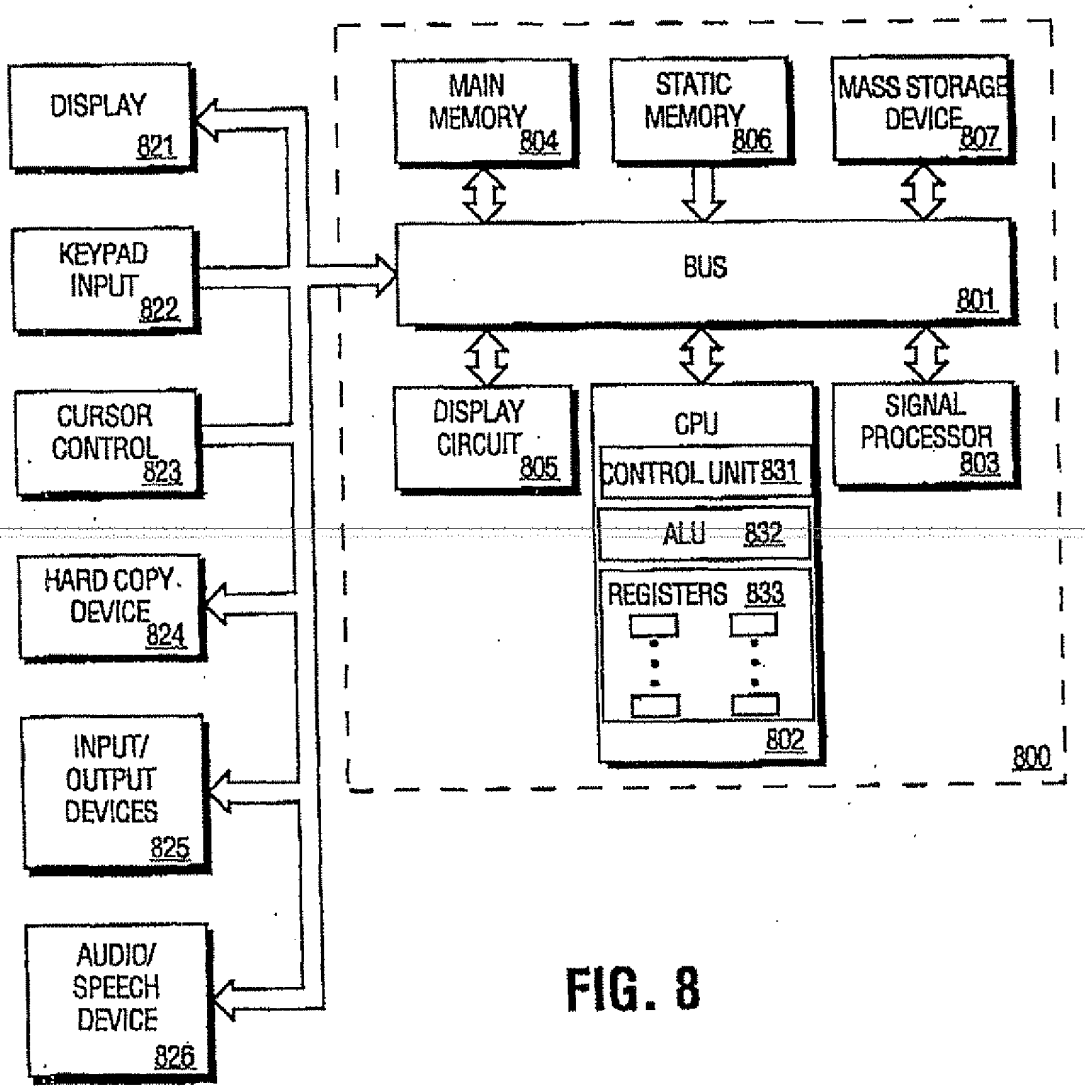

DISPLAY
821
KEYPAD
INPUT
822
CURSOR
CONTROL
823
HARD COPY
DEVICE
824
INPUT/
OUTPUT
DEVICES
825
AUDIO/
SPEECH
DEVICE
826
MAIN
MEMORY
804
STATIC
MEMORY
806
MASS STORAGE
DEVICE
807
BUS
801
DISPLAY
CIRCUIT
805
CPU
CONTROL UNIT 831
ALU 832
REGISTERS 833
802
SIGNAL
PROCESSOR
803
800

FIG. 8

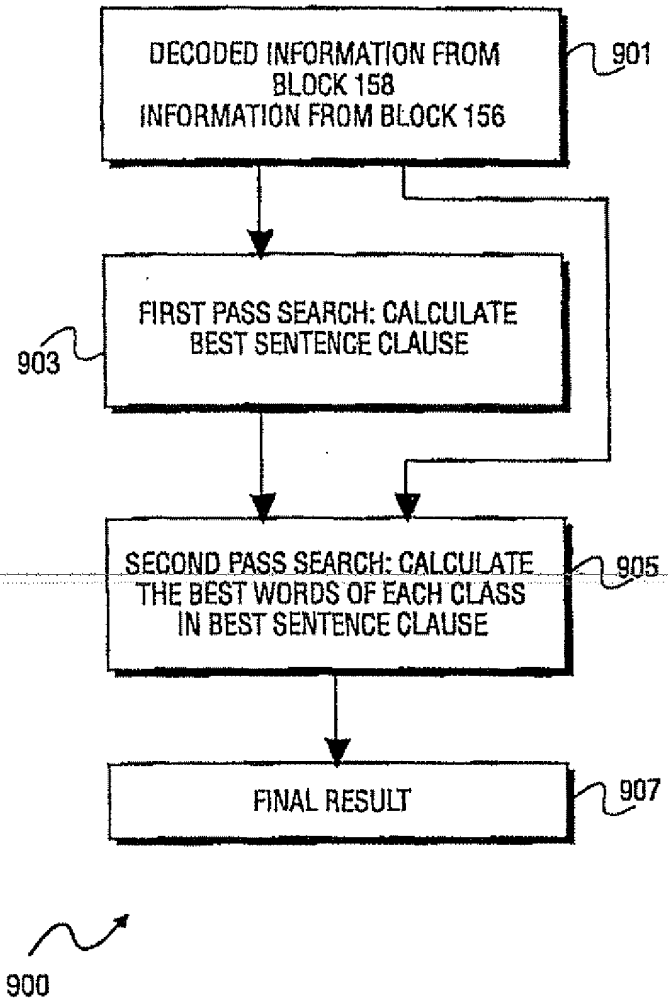

DECODED INFORMATION FROM BLOCK 158
INFORMATION FROM BLOCK 156
901
FIRST PASS SEARCH: CALCULATE BEST SENTENCE CLAUSE
903
SECOND PASS SEARCH: CALCULATE THE BEST WORDS OF EACH CLASS IN BEST SENTENCE CLAUSE
905
FINAL RESULT
907
900

FIG. 9

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN00/00297

A. CLASSIFICATION OF SUBJECT MATTER

G10L15/14

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

chinese patents

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, EPODOC, WPI, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP-A-0938076, (APPLE COMPUTER,INC.) 25 Aug 1999(25.08.99)	1-14
A	US-A-5752052, (MICROSOFT CORPORATION) 12 May 1998(12.05.98)	1-27
A	WO-A-9637881, (APPLIED LANGUAGE TECHNOLOGIES) 28 Nov 1996(28.11.96)	1-14

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim (S) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 Mar 2001(22.03.01)	29 MAR 2001 (29.03.01)
Name and mailing address of the ISA/CN 6 Xitucheng Rd., Jimen Bridge, Haidian District, 100088 Beijing, China Facsimile No. 86-10-62019451	Authorized officer YU, Zhilong Telephone No. +86-10-62093918

Form PCT/ISA /210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN00/00297

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP-A-0938076	25-08-1999	CA-A-2151370	21-07-1994
		WO-A-9416435	21-07-1994
		AU-A-5962494	15-08-1994
		US-A-5384892	24-01-1995
		US-A-5390279	14-02-1995
		EP-A-0677203	18-10-1995
		JP-T-8505957	25-06-1996
		US-A-5613036	18-03-1997
		EP-A-0938077	25-08-1999
WO-A-9637881	28-11-1996	CA-A-2220004	28-11-1996
		AU-A-5738296	11-12-1996
		EP-A-0838073	29-04-1998
US-A-5752052	12-05-1998	WO-A-9600436	04-01-1996
		EP-A-0715756	12-06-1996
		JP-T-9505173	20-05-1997
		US-A-5963894	05-10-1999